Design of Data Analysis

March, 2023

Table of Contents

# About this Course

The concept of reproducibility of a scientific investigation is most directly connected to the analysis of the data. Therefore, this course provides an overview of the structure of a data analysis and how to design an analysis to maximize the potential for reproducibility by research team members and by others. A key premise of this course is that reproducibility failure is often a consequence of poorly specified data analyses and a lack of understanding of the primary goal of the analysis by key stakeholders. The idea that “prevention is the best medicine” underlies this module, as proper design, structure, and execution of data analyses can prevent problems further downstream. We will therefore cover key analytic design concepts such as analytic iteration, analysis requirements and expectation setting, and unexpected outcomes and root causes.

## Learning Objectives

1. Implement the basic analytic cycle of the sense-making process: setting expectations about the data, comparing results to expectations, diagnosing unexpected outcomes, hypothesizing explanations and alternatives, and revising our understanding of the data.
2. Given a problem statement, characterize the stakeholders for a data analysis and what they hope to obtain/learn from the analysis
3. Write a high-level description of final/key/primary outputs and measures of uncertainty to be produced by the analysis
4. Develop a summary of key assumptions or constraints concerning inputs, data, or other factors.
5. Develop a set of requirements/expectations for the analysis regarding the outputs
6. Identify key analytic and data processing choices in an analysis that may significantly affect outputs
7. Identify any unexpected outcomes for a given analysis plan
8. Apply techniques for identifying root causes of any unexpected outcomes produced by the analysis

## Available course formats

This course is available in multiple formats which allows you to take it in the way that best suites your needs. You can take it for certificate which can be for free or fee.

* The material for this course can be viewed without login requirement on this [Bookdown website](http://hutchdatascience.org/Design_of_Data_Analysis/). This format might be most appropriate for you if you rely on screen-reader technology.
* Our courses are open source, you can find the [source material for this course on GitHub](https://github.com/fhdsl/Design_of_Data_Analysis/).

# 1 Analytic Iteration

## Learning Objectives

The learning objectives for this chapter are:

1. Describe the basic analytic cycle of the sense-making process.
2. Describe the importance of setting expectations about the data, comparing results to expectations, diagnosing unexpected outcomes, hypothesizing explanations and alternatives, and revising our understanding of the data.

## 1.1 Introduction

To the uninitiated, a data analysis may appear to follow a linear, one-step-after-the-other process which at the end, arrives at a nicely packaged and coherent result. In reality, data analysis is a highly iterative and non-linear process, better reflected by a series of epicycles (see Figure), in which information is learned at each step, which then informs whether (and how) to refine, and redo, the step that was just performed, or whether (and how) to proceed to the next step.

An epicycle is a small circle whose center moves around the circumference of a larger circle. In data analysis, the iterative process that is applied to all steps of the data analysis can be conceived of as an epicycle that is repeated for each step along the circumference of the entire data analysis process. Some data analyses appear to be fixed and linear, such as algorithms embedded into various software platforms, including apps. However, these algorithms are final data analysis products that have emerged from the very non-linear work of developing and refining a data analysis so that it can be “algorithmized.”

Before diving into the “epicycle of analysis,” it’s helpful to pause and consider what we mean by a “data analysis.” Although many of the concepts we will discuss in this book are applicable to conducting a *study*, the framework and concepts in this, and subsequent, chapters are tailored specifically to conducting a *data analysis*. While a study includes developing and executing a plan for collecting data, a data analysis presumes the data have already been collected. More specifically, a study includes the development of a hypothesis or question, the designing of the data collection process (or study protocol), the collection of the data, and the analysis and interpretation of the data. Because a data analysis presumes that the data have already been collected, it includes development and refinement of a question and the process of analyzing and interpreting the data. It is important to note that although a data analysis is often performed without conducting a study, it may also be performed as a component of a study.

## 1.2 Epicycle of Analysis

There are 5 core activities of data analysis:

1. Stating and refining the question
2. Exploring the data
3. Building formal statistical models
4. Interpreting the results
5. Communicating the results

These 5 activities can occur at different time scales: for example, you might go through all 5 in the course of a day, but also deal with each, for a large project, over the course of many months. Before discussing these core activities, which will occur in later chapters, it will be important to first understand the overall framework used to approach each of these activities.

Although there are many different types of activities that you might engage in while doing data analysis, every aspect of the entire process can be approached through an iterative process that we call the “epicycle of data analysis”. More specifically, for each of the five core activities, it is critical that you engage in the following steps:

1. Setting Expectations,
2. Collecting information (data), comparing the data to your expectations, and if the expectations don’t match,
3. Revising your expectations or fixing the data so your data and your expectations match.

Iterating through this 3-step process is what we call the “epicycle of data analysis.” As you go through every stage of an analysis, you will need to go through the epicycle to continuously refine your question, your exploratory data analysis, your formal models, your interpretation, and your communication.

The repeated cycling through each of these five core activities that is done to complete a data analysis forms the larger circle of data analysis (See Figure). In this chapter we go into detail about what this 3-step epicyclic process is and give examples of how you can apply it to your data analysis.

Epicycles of Analysis

Figure : Epicycles of Analysis

## 1.3 Setting Expectations

Developing expectations is the process of deliberately thinking about what you expect before you do anything, such as inspect your data, perform a procedure, or enter a command. For experienced data analysts, in some circumstances, developing expectations may be an automatic, almost subconscious process, but it’s an important activity to cultivate and be deliberate about.

For example, you may be going out to dinner with friends at a cash-only establishment and need to stop by the ATM to withdraw money before meeting up. To make a decision about the amount of money you’re going to withdraw, you have to have developed some expectation of the cost of dinner. This may be an automatic expectation because you dine at this establishment regularly so you know what the typical cost of a meal is there, which would be an example of *a priori* knowledge. Another example of *a priori* knowledge would be knowing what a typical meal costs at a restaurant in your city, or knowing what a meal at the most expensive restaurants in your city costs. Using that information, you could perhaps place an upper and lower bound on how much the meal will cost.

You may have also sought out external information to develop your expectations, which could include asking your friends who will be joining you or who have eaten at the restaurant before and/or Googling the restaurant to find general cost information online or a menu with prices. This same process, in which you use any a priori information you have and/or external sources to determine what you expect when you inspect your data or execute an analysis procedure, applies to each core activity of the data analysis process.

## 1.4 Collecting Information

This step entails collecting information about your question or your data. For your question, you collect information by performing a literature search or asking experts in order to ensure that your question is a good one. In the next chapter, we will discuss characteristics of a good question. For your data, after you have some expectations about what the result will be when you inspect your data or perform the analysis procedure, you then perform the operation. The results of that operation are the data you need to collect, and then you determine if the data you collected matches your expectations. To extend the restaurant metaphor, when you go to the restaurant, getting the check is collecting the data.

## 1.5 Comparing Expectations to Data

Now that you have data in hand (the check at the restaurant), the next step is to compare your expectations to the data. There are two possible outcomes: either your expectations of the cost match the amount on the check, or they do not. If your expectations and the data match, terrific, you can move onto the next activity. If, on the other hand, your expectations were a cost of 30 dollars, but the check was 40 dollars, your expectations and the data do not match. There are two possible explanations for the discordance: first, your expectations were wrong and need to be revised, or second, the check was wrong and contains an error. You review the check and find that you were charged for two desserts instead of the one that you had, and conclude that there is an error in the data, so ask for the check to be corrected.

One key indicator of how well your data analysis is going is how easy or difficult it is to match the data you collected to your original expectations. You want to setup your expectations and your data so that matching the two up is easy. In the restaurant example, your expectation was $30 and the data said the meal cost $40, so it’s easy to see that (a) your expectation was off by $10 and that (b) the meal was more expensive than you thought. When you come back to this place, you might bring an extra $10. If our original expectation was that the meal would be between $0 and $1,000, then it’s true that our data fall into that range, but it’s not clear how much more we’ve learned. For example, would you change your behavior the next time you came back? The expectation of a $30 meal is sometimes referred to as a sharp hypothesis because it states something very specific that can be verified with the data.

## 1.6 Applying the Epicycle of Analysis Process

Before we discuss a couple of examples, let’s review the three steps to use for each core data analysis activity. These are :

1. Setting expectations,
2. Collecting information (data), comparing the data to your expectations, and if the expectations don’t match,
3. Revising your expectations or fixing the data so that your expectations and the data match.

### 1.6.1 Example: Asthma prevalence in the U.S.

Let’s apply the “data analysis epicycle” to a very basic example. Let’s say your initial question is to determine the prevalence of asthma among adults, because your company wants to understand how big the market might be for a new asthma drug. You have a general question that has been identified by your boss, but need to: (1) sharpen the question, (2) explore the data, (3) build a statistical model, (4) interpret the results, and (5) communicate the results. We’ll apply the “epicycle” to each of these five core activities.

For the first activity, refining the question, you would first develop your expectations of the question, then collect information about the question and determine if the information you collect matches your expectations, and if not, you would revise the question. Your expectations are that the answer to this question is unknown and that the question is answerable. A literature and internet search, however, reveal that this question has been answered (and is continually answered by the Centers for Disease Control (CDC)), so you reconsider the question since you can simply go to the CDC website to get recent asthma prevalence data.

You inform your boss and initiate a conversation that reveals that any new drug that was developed would target those whose asthma was not controlled with currently available medication, so you identify a better question, which is “how many people in the United States have asthma that is not currently controlled, and what are the demographic predictors of uncontrolled asthma?” You repeat the process of collecting information to determine if your question is answerable and is a good one, and continue this process until you are satisfied that you have refined your question so that you have a good question that can be answered with available data.

Let’s assume that you have identified a data source that can be downloaded from a website and is a sample that represents the United States adult population, 18 years and older. The next activity is exploratory data analysis, and you start with the expectation that when you inspect your data that there will be 10,123 rows (or records), each representing an individual in the US as this is the information provided in the documentation, or codebook, that comes with the dataset. The codebook also tells you that there will be a variable indicating the age of each individual in the dataset.

When you inspect the data, though, you notice that there are only 4,803 rows, so return to the codebook to confirm that your expectations are correct about the number of rows, and when you confirm that your expectations are correct, you return to the website where you downloaded the files and discover that there were two files that contained the data you needed, with one file containing 4,803 records and the second file containing the remaining 5,320 records. You download the second file and read it into your statistical software package and append the second file to the first.

Now you have the correct number of rows, so you move on to determine if your expectations about the age of the population matches your expectations, which is that everyone is 18 years or older. You summarize the age variable, so you can view the minimum and maximum values and find that all individuals are 18 years or older, which matches your expectations. Although there is more that you would do to inspect and explore your data, these two tasks are examples of the approach to take. Ultimately, you will use this data set to estimate the prevalence of uncontrolled asthma among adults in the US.

The third activity is building a statistical model, which is needed in order to determine the demographic characteristics that best predict that someone has uncontrolled asthma. Statistical models serve to produce a precise formulation of your question so that you can see exactly how you want to use your data, whether it is to estimate a specific parameter or to make a prediction. Statistical models also provide a formal framework in which you can challenge your findings and test your assumptions.

Now that you have estimated the prevalence of uncontrolled asthma among US adults and determined that age, gender, race, body mass index, smoking status, and income are the best predictors of uncontrolled asthma available, you move to the fourth core activity, which is interpreting the results. In reality, interpreting results happens along with model building as well as after you’ve finished building your model, but conceptually they are distinct activities.

Let’s assume you’ve built your final model and so you are moving on to interpreting the findings of your model. When you examine your final predictive model, initially your expectations are matched as age, African American/black race, body mass index, smoking status, and low income are all positively associated with uncontrolled asthma.

However, you notice that female gender is *inversely* associated with uncontrolled asthma, when your research and discussions with experts indicate that among adults, female gender should be positively associated with uncontrolled asthma. This mismatch between expectations and results leads you to pause and do some exploring to determine if your results are indeed correct and you need to adjust your expectations or if there is a problem with your results rather than your expectations. After some digging, you discover that you had thought that the gender variable was coded 1 for female and 0 for male, but instead the codebook indicates that the gender variable was coded 1 for male and 0 for female. So the interpretation of your results was incorrect, not your expectations. Now that you understand what the coding is for the gender variable, your interpretation of the model results matches your expectations, so you can move on to communicating your findings.

Lastly, you communicate your findings, and yes, the epicycle applies to communication as well. For the purposes of this example, let’s assume you’ve put together an informal report that includes a brief summary of your findings. Your expectation is that your report will communicate the information your boss is interested in knowing. You meet with your boss to review the findings and she asks two questions: (1) how recently the data in the dataset were collected and (2) how changing demographic patterns projected to occur in the next 5-10 years would be expected to affect the prevalence of uncontrolled asthma. Although it may be disappointing that your report does not fully meet your boss’s needs, getting feedback is a critical part of doing a data analysis, and in fact, we would argue that a good data analysis requires communication, feedback, and then actions in response to the feedback.

Although you know the answer about the years when the data were collected, you realize you did not include this information in your report, so you revise the report to include it. You also realize that your boss’s question about the effect of changing demographics on the prevalence of uncontrolled asthma is a good one since your company wants to predict the size of the market in the future, so you now have a new data analysis to tackle. You should also feel good that your data analysis brought additional questions to the forefront, as this is one characteristic of a successful data analysis.

In the next chapters, we will make extensive use of this framework to discuss how each activity in the data analysis process needs to be continuously iterated. While executing the three steps may seem tedious at first, eventually, you will get the hang of it and the cycling of the process will occur naturally and subconsciously. Indeed, we would argue that most of the best data analysts don’t even realize they are doing this!

# 2 Analytic Design

## Learning Objectives

The learning objectives of this chapter are:

1. Describe the six types of questions that we might ask in a data analysis
2. Describe the characteristics of a good question

## 2.1 Types of Questions

Doing data analysis requires quite a bit of thinking and we believe that when you’ve completed a good data analysis, you’ve spent more time thinking than doing. The thinking begins before you even look at a dataset, and it’s well worth devoting careful thought to your question. This point cannot be over-emphasized as many of the “fatal” pitfalls of a data analysis can be avoided by expending the mental energy to get your question right. In this chapter, we will discuss the characteristics of a good question, the types of questions that can be asked, and how to apply the iterative epicyclic process to stating and refining your question so that when you start looking at data, you have a sharp, answerable question.

Before we delve into stating the question, it’s helpful to consider what the different types of questions are. There are six basic types of questions and much of the discussion that follows comes from a [paper](http://www.sciencemag.org/content/347/6228/1314.short) published in *Science* by Roger and [Jeff Leek](http://jtleek.com). Understanding the type of question you are asking may be the most fundamental step you can take to ensure that, in the end, your interpretation of the results is correct. The six types of questions are:

1. Descriptive
2. Exploratory
3. Inferential
4. Predictive
5. Causal
6. Mechanistic

And the type of question you are asking directly informs how you interpret your results.

A *descriptive* question is one that seeks to summarize a characteristic of a set of data. Examples include determining the proportion of males, the mean number of servings of fresh fruits and vegetables per day, or the frequency of viral illnesses in a set of data collected from a group of individuals. There is no interpretation of the result itself as the result is a fact, an attribute of the set of data that you are working with.

An *exploratory* question is one in which you analyze the data to see if there are patterns, trends, or relationships between variables. These types of analyses are also called “hypothesis-generating” analyses because rather than testing a hypothesis as would be done with an inferential, causal, or mechanistic question, you are looking for patterns that would support proposing a hypothesis. If you had a general thought that diet was linked somehow to viral illnesses, you might explore this idea by examining relationships between a range of dietary factors and viral illnesses. You find in your exploratory analysis that individuals who ate a diet high in certain foods had fewer viral illnesses than those whose diet was not enriched for these foods, so you propose the hypothesis that among adults, eating at least 5 servings a day of fresh fruit and vegetables is associated with fewer viral illnesses per year.

An *inferential* question would be a restatement of this proposed hypothesis as a question and would be answered by analyzing a different set of data, which in this example, is a representative sample of adults in the US. By analyzing this different set of data you are both determining if the association you observed in your exploratory analysis holds in a different sample and whether it holds in a sample that is representative of the adult US population, which would suggest that the association is applicable to all adults in the US. In other words, you will be able to infer what is true, on average, for the adult population in the US from the analysis you perform on the representative sample.

A *predictive* question would be one where you ask what types of people will eat a diet high in fresh fruits and vegetables during the next year. In this type of question you are less interested in what causes someone to eat a certain diet, just what predicts whether someone will eat this certain diet. For example, higher income may be one of the final set of predictors, and you may not know (or even care) why people with higher incomes are more likely to eat a diet high in fresh fruits and vegetables, but what is most important is that income is a factor that predicts this behavior.

Although an inferential question might tell us that people who eat a certain type of foods tend to have fewer viral illnesses, the answer to this question does not tell us if eating these foods causes a reduction in the number of viral illnesses, which would be the case for a *causal* question. A causal question asks about whether changing one factor will change another factor, on average, in a population. Sometimes the underlying design of the data collection, by default, allows for the question that you ask to be causal. An example of this would be data collected in the context of a randomized trial, in which people were randomly assigned to eat a diet high in fresh fruits and vegetables or one that was low in fresh fruits and vegetables. In other instances, even if your data are not from a randomized trial, you can take an analytic approach designed to answer a causal question.

Finally, none of the questions described so far will lead to an answer that will tell us, if the diet does, indeed, cause a reduction in the number of viral illnesses, *how* the diet leads to a reduction in the number of viral illnesses. A question that asks how a diet high in fresh fruits and vegetables leads to a reduction in the number of viral illnesses would be a *mechanistic* question.

There are a couple of additional points about the types of questions that are important. First, by necessity, many data analyses answer multiple types of questions. For example, if a data analysis aims to answer an inferential question, descriptive and exploratory questions must also be answered during the process of answering the inferential question. To continue our example of diet and viral illnesses, you would not jump straight to a statistical model of the relationship between a diet high in fresh fruits and vegetables and the number of viral illnesses without having determined the frequency of this type of diet and viral illnesses and their relationship to one another in this sample. A second point is that the type of question you ask is determined in part by the data available to you (unless you plan to conduct a study and collect the data needed to do the analysis). For example, you may want to ask a causal question about diet and viral illnesses to know whether eating a diet high in fresh fruits and vegetables causes a decrease in the number of viral illnesses, and the best type of data to answer this causal question is one in which people’s diets change from one that is high in fresh fruits and vegetables to one that is not, or vice versa. If this type of data set does not exist, then the best you may be able to do is either apply causal analysis methods to observational data or instead answer an inferential question about diet and viral illnesses.

## 2.2 Applying the Epicycle to Stating and Refining Your Question

You can now use the information about the types of questions and characteristics of good questions as a guide to refining your question. To accomplish this, you can iterate through the 3 steps of:

1. Establishing your expectations about the question
2. Gathering information about your question
3. Determining if your expectations match the information you gathered, and then refining your question (or expectations) if your expectations did not match the information you gathered

## 2.3 Characteristics of a Good Question

There are five key characteristics of a good question for a data analysis, which range from the very basic characteristic that the question should not have already been answered to the more abstract characteristic that each of the possible answers to the question should have a single interpretation and be meaningful. We will discuss how to assess this in greater detail below.

As a start, the question should be of **interest** to your audience, the identity of which will depend on the context and environment in which you are working with data. If you are in academia, the audience may be your collaborators, the scientific community, government regulators, your funders, and/or the public. If you are working at a start-up, your audience is your boss, the company leadership, and the investors. As an example, answering the question of whether outdoor particulate matter pollution is associated with developmental problems in children may be of interest to people involved in regulating air pollution, but may not be of interest to a grocery store chain. On the other hand, answering the question of whether sales of pepperoni are higher when it is displayed next to the pizza sauce and pizza crust or when it is displayed with the other packaged meats would be of interest to a grocery store chain, but not to people in other industries.

You should also check that the question has **not already been answered.** With the recent explosion of data, the growing amount of publicly available data, and the seemingly endless scientific literature and other resources, it is not uncommon to discover that your question of interest has been answered already. Some research and discussion with experts can help sort this out, and can also be helpful because even if the specific question you have in mind has not been answered, related questions may have been answered and the answers to these related questions are informative for deciding if or how you proceed with your specific question.

The question should also stem from a **plausible** framework. In other words, the question above about the relationship between sales of pepperoni and its placement in the store is a plausible one because shoppers buying pizza ingredients are more likely than other shoppers to be interested in pepperoni and may be more likely to buy it if they see it at the same time that they are selecting the other pizza ingredients. A less plausible question would be whether pepperoni sales correlate with yogurt sales, unless you had some prior knowledge suggesting that these should be correlated.

If you ask a question whose framework is not plausible, you are likely to end up with an answer that’s difficult to interpret or have confidence in. In the pepperoni-yogurt question, if you do find they are correlated, many questions are raised about the result itself: is it really correct?, why are these things correlated- is there another explanation?, and others. You can ensure that your question is grounded in a plausible framework by using your own knowledge of the subject area and doing a little research, which together can go a long way in terms of helping you sort out whether your question is grounded in a plausible framework.

The question, should also, of course, be **answerable**. Although perhaps this doesn’t need stating, it’s worth pointing out that some of the best questions aren’t answerable - either because the data don’t exist or there is no means of collecting the data because of lack of resources, feasibility, or ethical problems. For example, it is quite plausible that there are defects in the functioning of certain cells in the brain that cause autism, but it not possible to perform brain biopsies to collect live cells to study, which would be needed to answer this question.

**Specificity** is also an important characteristic of a good question. An example of a general question is: Is eating a healthier diet better for you? Working towards specificity will refine your question and directly inform what steps to take when you start looking at data. A more specific question emerges after asking yourself what you mean by a “healthier” diet and when you say something is “better for you”? The process of increasing the specificity should lead to a final, refined question such as: “Does eating at least 5 servings per day of fresh fruits and vegetables lead to fewer upper respiratory tract infections (colds)?” With this degree of specificity, your plan of attack is much clearer and the answer you will get at the end of the data analysis will be more interpretable as you will either recommend or not recommend the specific action of eating at least 5 servings of fresh fruit and vegetables per day as a means of protecting against upper respiratory tract infections.

## 2.4 Translating a Question into a Data Problem

Another aspect to consider when you’re developing your question is what will happen when you translate it into a data problem. Every question must be operationalized as a data analysis that leads to a result. Pausing to think through what the results of the data analysis would look like and how they might be interpreted is important as it can prevent you from wasting a lot of time embarking on an analysis whose result is not interpretable. Although we will discuss many examples of questions that lead to interpretable and meaningful results throughout the book, it may be easiest to start first by thinking about what sorts of questions *don’t* lead to interpretable answers.

The typical type of question that does not meet this criterion is a question that uses inappropriate data. For example, your question may be whether taking a vitamin D supplement is associated with fewer headaches, and you plan on answering that question by using the number of times a person took a pain reliever as a marker of the number of headaches they had. You may find an association between taking vitamin D supplements and taking less pain reliever medication, but it won’t be clear what the interpretation of this result is. In fact, it is possible that people who take vitamin D supplements also tend to be less likely to take other over-the-counter medications just because they are “medication avoidant,” and not because they are actually getting fewer headaches. It may also be that they are using less pain reliever medication because they have less joint pain, or other types of pain, but not fewer headaches. Another interpretation, of course, is that they are indeed having fewer headaches, but the problem is that you can’t determine whether this is the correct interpretation or one of the other interpretations is correct. In essence, the problem with this question is that for a single possible answer, there are multiple interpretations. This scenario of multiple interpretations arises when at least one of the variables you use (in this case, pain reliever use) is not a good measure of the concept you are truly after (in this case, headaches). To head off this problem, you will want to make sure that the data available to answer your question provide reasonably specific measures of the factors required to answer your question.

A related problem that interferes with interpretation of results is confounding. Confounding is a potential problem when your question asks about the relationship between factors, such as taking vitamin D and frequency of headaches. A brief description of the concept of confounding is that it is present when a factor that you were not necessarily considering in your question is related to both your exposure of interest (in the example, taking vitamin D supplements) and your outcome of interest (taking pain reliever medication). For example, income could be a confounder, because it may be related to both taking vitamin D supplements and frequency of headaches, since people with higher income may tend to be more likely to take a supplement and less likely to have chronic health problems, such as headaches. Generally, as long as you have income data available to you, you will be able to adjust for this confounder and reduce the number of possible interpretations of the answer to your question. As you refine your question, spend some time identifying the potential confounders and thinking about whether your dataset includes information about these potential confounders.

Another type of problem that can occur when inappropriate data are used is that the result is not interpretable because the underlying way in which the data were collected lead to a biased result. For example, imagine that you are using a dataset created from a survey of women who had had children. The survey includes information about whether their children had autism and whether they reported eating sushi while pregnant, and you see an association between report of eating sushi during pregnancy and having a child with autism. However, because women who have had a child with a health condition recall the exposures, such as raw fish, that occurred during pregnancy differently than those who have had healthy children, the observed association between sushi exposure and autism may just be the manifestation of a mother’s tendency to focus more events during pregnancy when she has a child with a health condition. This is an example of recall bias, but there are many types of bias that can occur.

The other major bias to understand and consider when refining your question is selection bias, which occurs when the data your are analyzing were collected in such a way to inflate the proportion of people who have both characteristics above what exists in the general population. If a study advertised that it was a study about autism and diet during pregnancy, then it is quite possible that women who both ate raw fish and had a child with autism would be more likely to respond to the survey than those who had one of these conditions or neither of these conditions. This scenario would lead to a biased answer to your question about mothers’ sushi intakes during pregnancy and risk of autism in their children. A good rule of thumb is that if you are examining relationships between two factors, bias may be a problem if you are more (or less) likely to observe individuals with both factors because of how the population was selected, or how a person might recall the past when responding to a survey. There will be more discussion about bias in subsequent chapters on ([Inference: A Primer](#chapter-inference) and [Interpreting Your Results](#chapter-interpretation)), but the best time to consider its effects on your data analysis is when you are identifying the question you will answer and thinking about how you are going to answer the question with the data available to you.

## 2.5 Case Study

Joe works for a company that makes a variety of fitness tracking devices and apps and the name of the company is Fit on Fleek. Fit on Fleek’s goal is, like many tech start-ups, to use the data they collect from users of their devices to do targeted marketing of various products. The product that they would like to market is a new one that they have just developed and not yet started selling, which is a sleep tracker and app that tracks various phases of sleep, such as REM sleep, and also provides advice for improving sleep. The sleep tracker is called Sleep on Fleek.

Joe’s boss asks him to analyze the data that the company has on its users of their health tracking devices and apps to identify users for targeted Sleep on Fleek ads. Fit on Fleek has the following data from each of their customers: basic demographic information, number of steps walked per day, number of flights of stairs climbed per day, sedentary awake hours per day, hours of alertness per day, hours of drowsiness per day, and hours slept per day (but not more detailed information about sleep that the sleep tracker would track).

Although Joe has an objective in mind, gleaned from a discussion with his boss, and he also knows what types of data are available in the Fit on Fleek database, he does not yet have a question. This scenario, in which Joe is given an objective, but not a question, is common, so Joe’s first task is to translate the objective into a question, and this will take some back-and-forth communication with his boss. The approach to informal communications that take place during the process of the data analysis project, is covered in detail in the [Communication chapter](#chapter-communication). After a few discussions, Joe settles on the following question: “Which Fit on Fleek users don’t get enough sleep?” He and his boss agree that the customers who would be most likely to be interested in purchasing the Sleep on Fleek device and app are those who appear to have problems with sleep, and the easiest problem to track and probably the most common problem is not getting enough sleep.

You might think that since Joe now has a question, that he should move to download the data and start doing exploratory analyses, but there is a bit of work Joe still has to do to refine the question. The two main tasks Joe needs to tackle are: (1) to think through how his question does, or does not, meet the characteristics of a good question and (2) to determine what type of question he is asking so that he has a good understanding of what kinds of conclusions can (and cannot) be drawn when he has finished the data analysis.

Joe reviews the characteristics of a good question and his expectations are that his question has all of these characteristics: -of interest -not already answered -grounded in a plausible framework -answerable -specific

The answer that he will get at the end of his analysis (when he translates his question into a data problem) should also be interpretable.

He then thinks through what he knows about the question and in his judgment, the question is of interest as his boss expressed interest.

He also knows that the question could not have been answered already since his boss indicated that it had not and a review of the company’s previous data analyses reveals no previous analysis designed to answer the question.

Next he assesses whether the question is grounded in a plausible framework. The question, “Which Fit on Fleek users don’t get enough sleep?”, seems to be grounded in plausibility as it makes sense that people who get too little sleep would be interested in trying to improve their sleep by tracking it. However, Joe wonders whether the duration of sleep is the best marker for whether a person feels that they are getting inadequate sleep. He knows some people who regularly get little more than 5 hours of sleep a night and they seem satisfied with their sleep. Joe reaches out to a sleep medicine specialist and learns that a better measure of whether someone is affected by lack of sleep or poor quality sleep is daytime drowsiness. It turns out that his initial expectation that the question was grounded in a plausible framework did not match the information he received when he spoke with a content expert. So he revises his question so that it matches his expectations of plausibility and the revised question is: Which Fit on Fleek users have drowsiness during the day?

Joe pauses to make sure that this question is, indeed, answerable with the data he has available to him, and confirms that it is. He also pauses to think about the specificity of the question. He believes that it is specific, but goes through the exercise of discussing the question with colleagues to gather information about the specificity of the question. When he raises the idea of answering this question, his colleagues ask him many questions about what various parts of the question mean: what is meant by “which users”? Does this mean: What are the demographic characteristics of the users who have drowsiness? Or something else? What about “drowsiness during the day”? Should this phrase mean any drowsiness on any day? Or drowsiness lasting at least a certain amount of time on at least a certain number of days? The conversation with colleagues was very informative and indicated that the question was not very specific. Joe revises his question so that it is now specific: “Which demographic and health characteristics identify users who are most likely to have chronic drowsiness, defined as at least one episode of drowsiness at least every other day?”

Joe now moves on to thinking about what the possible answers to his questions are, and whether they will be interpretable. Joe identifies two possible outcomes of his analysis: (1) there are no characteristics that identify people who have chronic daytime drowsiness or (2) there are one or more characteristics that identify people with chronic daytime drowsiness. These two possibilities are interpretable and meaningful. For the first, Joe would conclude that targeting ads for the Sleep on Fleek tracker to people who are predicted to have chronic daytime drowsiness would not be possible, and for the second, he’d conclude that targeting the ad is possible, and he’d know which characteristic(s) to use to select people for the targeted ads.

Now that Joe has a good question in hand, after iterating through the 3 steps of the epicycle as he considered whether his question met each of the characteristics of a good question, the next step is for him to figure out what type of question he has. He goes through a thought process similar to the process he used for each of the characteristics above. He starts thinking that his question is an exploratory one, but as he reviews the description and examples of an exploratory question, he realizes that although some parts of the analysis he will do to answer the question will be exploratory, ultimately his question is more than exploratory because its answer will predict which users are likely to have chronic daytime drowsiness, so his question is a prediction question. Identifying the type of question is very helpful because, along with a good question, he now knows that he needs to use a prediction approach in his analyses, in particular in the model-building phase (see [Formal Modeling chapter](#chapter-formal)).

## 2.6 Concluding Thoughts

By now, you should be poised to apply the 3 steps of the epicycle to stating and refining a question. If you are a seasoned data analyst, much of this process may be automatic, so that you may not be entirely conscious of some parts of the process that lead you to a good question. Until you arrive at this point, this chapter can serve as a useful resource to you when you’re faced with the task of developing a good question. In the next chapters, we will discuss what to do with the data now that you have good question in hand.

# 3 Systems Thinking

## Learning Objectives

## 3.1 Introduction

The presentation of how data analyses are conducted is typically done in a forward manner. A question is posed, data are collected, and given the question and data, a system of statistical methods is assembled to produce evidence. That evidence is then interpreted in the context of the original question. While such a description provides a useful model, it is incomplete in that it assumes the statistical methods are completely determined by the question and the data. In practice, there is an equally important “backwards" process that data analysts use to either revise their statistical approach or investigate potential problems with the data. This process of revision is driven by observing deviations between the data and what we expect the data to look like.

Much previous work dedicated to studying the data analysis process has focused on the notion of “statistical thinking," or the cognitive processes that occur within the analyst while doing data analysis ([**wildpfannkuch1999?**](#ref-wildpfannkuch1999); [**grol:wick:2014?**](#ref-grol:wick:2014); [**wats:call:2003?**](#ref-wats:call:2003); [**hortonhardin2015?**](#ref-hortonhardin2015)). Grolemund and Wickham ([**grol:wick:2014?**](#ref-grol:wick:2014)) refer to these”forwards" and “backwards" aspects of data analysis as part of a sensemaking process, where analysts attempt to reconcile schema that describe the world with data that are measured from the world. Should there be a discrepancy between the schema and the data, the analyst can update the schema to better match the observed data. These updates ultimately result in knowledge being accumulated.

A key task for data analysts is to interpret the scientific question at hand, the available data, context, and resources, and assemble a system of statistical methods to analyze the data to produce a evidence. In general, statistical theory provides guidance on which methods should be applied in different data analytic scenarios. With knowledge of the context, scientific question, and the system of methods assembled, the data analyst can generate certain expectations for what the results of the analysis will be. However, once those methods are applied to the data and results are obtained, there is comparatively little formal methodology for reconciling differences between the observed results and our analytic expectations. In particular, knowledge of the cognitive process of statistical thinking does not suggest how to give feedback to analysts once the data are observed. While there is much previous work on data analysis assessments ([**Chance1997?**](#ref-Chance1997); [**Garfield2000?**](#ref-Garfield2000); [**Chance2002?**](#ref-Chance2002)) and project-based learning ([**Gnanadesikan1997?**](#ref-Gnanadesikan1997); [**Smith1998?**](#ref-Smith1998); [**Tishkovskaya2012?**](#ref-Tishkovskaya2012)), the literature provides little insight into how to recognize or formally assess why a data analysis has produced an unexpected outcome.

The question of how analysts should reconcile discrepancies between their expectations about the world and the observed data is the focus of this paper. We propose that a framework for characterizing this process emerges by considering the sequence of statistical methods being applied to the data as a *system* that induces a set of expected outcomes. This system then generates outputs that may deviate from our expectations and these deviations are referred to as *anomalies*. In a typical data analysis there may be many anomalies and the analyst must identify their root causes in order to determine the next step in the analysis.

Characterizing anomalies in data analysis requires that we have a thorough understanding of the entire system of statistical methods that are being applied to a given dataset. Included in this system are traditional statistical tools such as linear regression or hypothesis tests, as well as methods for reading data from their raw source, pre-processing, feature extraction, post-processing, and any final visualization tools that are applied to the model output. Ultimately, anomalies can be caused by any component of the system, not just the statistical model, and it is the data analyst’s job to understand the behavior of the entire system. Yet, there is little in the statistical literature that considers the complexity of such systems and how they might behave under real-world conditions ([**national1991future?**](#ref-national1991future)).

Shifting to a framework that focuses on anomalies in data analysis opens up new avenues for thinking about the data analysis process more generally. In particular, we can apply approaches from systems engineering to formally analyze how data analysts think about a problem. Tools like fault trees can provide a way of evaluating how well analysts understand the complex systems being applied to their data and can suggest improvements to those systems ([**vesely1981fault?**](#ref-vesely1981fault)). Comparing systems of statistical methods based on how they produce anomalies provides a new basis for choosing between approaches, in addition to traditional metrics such as efficiency or prediction accuracy. Finally, diagnosing data analytic anomalies provides a direct and generalizable approach to teaching data analysis on a large scale.

In this paper we introduce the concept of a statistical methods system, which is a collection of statistical tools that are applied to a dataset for answering a given question. We then discuss the notion of a set of expected outcomes and an anomaly space that is determined by the outputs of the system and the expected outcomes. Finally, we describe fault trees and how they can be used to analyze a statistical methods system with respect to different anomalies. We also provide some examples of how fault trees can be used to compare different statistical methods systems and choose between them.

## 3.2 Statistical Methods Systems

To study the manner in which data analyses can produced unexpected results, it is useful to first consider data analyses as a system of connected components that produce specific outputs. A *statistical methods system* is a collection of data analytic elements, procedures, and tools that are connected together to produce a data analysis *output*, such as a plot, summary statistic, parameter estimate, or other statistical quantity. By connecting these elements and tools together, we create a complex system through which data are transformed, summarized, visualized, and modeled ([**hick:peng:2019?**](#ref-hick:peng:2019); [**Breiman2001cultures?**](#ref-Breiman2001cultures)). Each of the components in the system will have its own inputs and outputs and tracing the path of those intermediate results plays a key role in developing an understanding of the system.

There are also contextual inputs to the data analysis, such as the main question or problem being addressed, the data, the choice of programming language to use, the audience, and the document or container for the analysis, such as Jupyter or R Notebooks. While these inputs are not necessarily decided or fundamentally modified by the analyst, the data analyst may be expected to provide feedback on some of these inputs. In particular, should an analysis produce an unexpected result, the analyst might identify one of these contextual inputs as the root cause of the problem.

A statistical methods system can be characterized as a sequence of steps or a deterministic algorithm. The algorithm is not necessarily uni-directional and may double-back on itself and branch into multiple sections. Ultimately, the algorithm produces an output that may be interpreted by the analyst or passed on to serve as input to another system. In describing the behavior of any system, one must be careful to define the resolution of the system (i.e. how detailed we want to specify steps) and the boundaries of the system diagram. In particular, we should acknowledge what elements are excluded from the diagram, such as application-specific context or other assumptions ([**vesely1981fault?**](#ref-vesely1981fault)). The development of a statistical methods system would typically be guided by statistical theory, as well as knowledge of the scientific question, any relevant context or previous research, available resources, and any design requirements for the system output.

### 3.2.1 Example: A Simple Linear Model System

Consider a system that fits a simple linear model as part of a data analysis. This system reads in some data on a pair of variables and , fits a simple linear regression via least squares, and outputs the intercept and slope estimates. A depiction of this system along with some representative R code is shown in Figure [1](#modelSLR).

Simple linear regression statistical methods system with pseudo-code.

Figure : Simple linear regression statistical methods system with pseudo-code.

The diagram indicates that understanding how this system operates requires knowledge of (1) how the data are read in; (2) how the model is fit to the data; and (3) how the estimated coefficients are extracted from the model fit and outputted. Specifically, if we are using R to analyze the data, we must have an understanding of the read.csv(), lm(), coef(), and print() functions.

## 3.3 Set of Expected Outcomes

Once a statistical methods system is built, but before it is applied to the data, we can develop a set of *expected outcomes* for the system. These expected outcomes represent our current state of scientific knowledge and may reflect any gaps or biases in our understanding of the data, the problem at hand, or the behavior of the statistical methods system. The overarching goal of the data analysis is to produce outputs that will in some way improve our understanding of the scientific problem. Without any expected outcomes, it is challenging to interpret the output of the system or determine how the output informs our understanding of the underlying data generation process.

An important property of the set of expected outcomes is that the expected outcomes are always stated in terms of the observed output of the system, *not* any underlying unobserved population parameters. We draw a distinction here between *hypotheses*, which are statements about the underlying population, and *expected outcomes*, which are statements about the observed sample data. Another property of the set of expected outcomes is that they will generally have sharp boundaries. Therefore, once we observe the output from the statistical methods system, we know immediately and with complete certainty whether the output is expected or unexpected. Boundaries of this nature are important for the analyst so that decisions can be made regarding any next steps in the analysis.

Developing a useful set of expected outcomes is part of the design process for a statistical methods system and depends on many factors, including our knowledge and assumptions about the underlying data generation process, our ability to model that process using statistical tools, our knowledge of the theoretical properties of those tools, and our understanding of the uncertainty or variability of the observed data across multiple samples. Hence, even though the underlying truth might be thought of as fixed, it is reasonable to assume that different analysts might develop different sets of expected outcomes, reflecting differing levels of familiarity with the various factors involved and different biases towards existing evidence or data.

### 3.3.1 Example: Expected Outcomes for the Sample Mean

In some contexts, the set of expected outcomes may be derived from formal statistical hypotheses. For example, we may design a system to compute the sample mean of a dataset that we model as being sampled independently from a distribution. In this case, the output from the system is and based on our current knowledge we may hypothesize that . Under that hypothesis, we might expect to fall between and . For , this interval is and any observed value of outside that interval would be an anomaly.

Another analyst might be more familiar with this data generation process and therefore hypothesize that the underlying population mean is without assuming a Normal distribution. This analyst might also know that the data collection process can be problematic, leading to very large observations on occasion. Therefore, based on experience and intuition, this analyst has a wider expected outcome interval of .

In both examples here, the set of expected outcomes was a statement about , the output of the system applied to the observed data. The set of expected outcomes was also a fixed interval with clear boundaries, making it straightforward to determine whether the output would fall in the interval or not.

## 3.4 Anomaly Space

An anomaly occurs only if there is a clearly defined set of expected outcomes for a system, and the observed output from the system does not fall into that set. The specification of an anomaly then requires three separate elements:

1. A description of *what* specific system output or collection of outputs is observed;
2. A description of *how* the outputs deviate from their expected outcomes; and
3. An indication of *when* or under what conditions the deviation is observed.

Continuing the example from Section [2.2.1](#sec:expectedmean) above, an anomaly for the sample mean could be “ is outside the expected interval of when a sample of size is inputted to the system”. The observed output is , the deviation is “outside the interval ", and the event occurs when .

The *anomaly space* of a statistical methods system consists of the collection of potential outputs from the system which would indicate that an anomaly has occurred. Fundamentally, the anomaly space is the complement of the set of expected outcomes. Not all areas of the anomaly space are equally important and in some applications it may be that anomalies occurring in certain subsets of the anomaly space are more interesting than anomalies occurring elsewhere. The size of the anomaly space of a statistical methods system is determined by the outputs produced by the system. Looking back to the simple linear model system in Figure [1](#modelSLR), there are only two outputs ( and ) that define the anomaly space. Therefore, any anomalies for that system must be determined by those two values.

As the number of system outputs grows, the size of the anomaly space may grow accordingly. For example, we can increase the size of the anomaly space for the simple linear model system by also returning the standard errors of the coefficients. With each additional output, we increase the number of ways in which anomalies can occur. Different systems with different sets of outputs will induce anomaly spaces of differing sizes and the nature of the anomaly space associated with a system may serve as a factor in choosing between systems. Also, because the anomaly space depends on the specification of the set of expected outcomes, different analysts with different expectations could induce different anomaly spaces for the same statistical methods system.

Once a statistical methods system has been applied to the data and an anomaly has been observed, the “forward" aspect of data analysis is complete and the analyst must begin the”backward" aspect to determine what if any changes should be made to the analysis. Such changes could involve modifying the statistical methods system itself or could require changes to our set of expected outcomes based on this new information. However, before any decision can be made in response to observing an anomaly, a data analyst must enumerate the possible *root causes* of the anomaly and determine which root causes should be investigated.

## 3.5 Fault Trees

The key formal tool that we introduce here is the *fault tree*, which can be used to analyze the design and operation of a statistical methods system and to identify root causes for any system anomalies. A fault tree is a tool commonly used in systems engineering for conducting a structured risk assessment and has a long history in aviation, aerospace, and nuclear power applications ([**vesely1981fault?**](#ref-vesely1981fault); [**michael2002fault?**](#ref-michael2002fault)). In general, fault trees can be used to discover the root cause of an anomaly after it occurs. However, an important use case for fault trees is to develop a comprehensive understanding of a system before an anomaly occurs. As such, it is a valuable design tool for complex systems.

A fault tree is a graphical tool that describes the possible combinations of causes and effects that lead to an anomaly. At the top of the tree is a description of an anomaly. The subsequent branches of the tree below the top event indicate possible causes of the event immediately above it in the tree. The tree can then be built recursively until we reach a root cause that cannot be further investigated. Each level of the tree is connected together using logic gates such as AND and OR gates. The leaf nodes of the tree indicate the root causes that may lead to an anomaly. Comprehensive details on constructing fault trees can be found in Vesely, et al. ([**vesely1981fault?**](#ref-vesely1981fault)) and Ericson ([**ericson2011fault?**](#ref-ericson2011fault)).

In principle, a fault tree can be constructed for every potential system anomaly that could be observed. However, if the goal is to develop an understanding of the strengths and weaknesses of a statistical methods system’s design, then it may be desirable to identify a few important anomalies whose fault trees provide substantial visibility into the operating characteristics of the system ([**vesely1981fault?**](#ref-vesely1981fault)).

## 3.6 Example: Fault Tree for a Simple Linear Model System

We can build a fault tree for the simple linear model system shown in Figure [1](#modelSLR). Under normal operation, we might expect that . With typical random variation in the data we might expect to range from 0 to about 3. Therefore, it would be unexpected to observe or . For this example, we will define the anomaly of interest as “ when outputted from the model fit". Note that although the set of expected outcomes is the interval , we define the anomaly as and ignore the part of the anomaly space defined by . Similarly, possible anomalies concerning the intercept will not be developed here.

Starting with the system description in Figure [1](#modelSLR), we can see that should we observe there are two possibilities we might consider: (1) the structural relationship between and has changed to no longer reflect the simple linear model; or (2) the underlying structural relationship remains, but the input data to the linear model has been contaminated, perhaps with outliers. Developing the “contaminated data" event a bit further, we can propose that either there are outliers present in the raw data or outliers were somehow introduced into the data before inputting to the regression model.

The completed fault tree is shown in Figure [2](#ftreeSLR) and was built using the FaultTree package in R ([**faulttreeRpackage2020?**](#ref-faulttreeRpackage2020)). The leaf nodes are labeled with circles to indicate the root cause events.

Fault tree for unexpected event of “Estimated coefficients outside of expected range when model fitted to the input data."

Figure : Fault tree for unexpected event of “Estimated coefficients outside of expected range when model fitted to the input data."

For a description of the notation and symbols used in the fault tree, see Appendix [7](#sec:notation). The fault tree indicates that outliers can be a root cause of the anomaly (events or ). However, outliers do not *always* cause an unexpected change in . The AND gate at indicates that the outliers also have to be arranged in such a manner that they cause to be , perhaps because of some selection process in the outlier generation ().

The other root cause is that there is a change in the underlying structural relationship between and or that we perhaps misunderstood the relationship between and in the first place. For example, it could be that the data are naturally more variable than we thought they were, and so our set of expected outcomes for should be larger. In any case, the AND gate at indicates that any structural change also needs to be large enough (relative to the noise in the data) so that we are able to observe it in the data.

### 3.6.1 Minimal Cutsets

A useful summary statistic from any properly constructed fault tree is the list of minimal cutsets, which is a summary of the minimal number of events that need to happen in order for the anomaly to occur. There may be many cutsets and there may be different cutsets with 1, 2, or more events in them. The fault tree in Figure [2](#ftreeSLR) has three cutsets of size 2: , , and . An example of one cutset is (1) There are outliers in the raw data (event ), AND (2) a selection process arranges the outliers such that they produce (event ). If this pair of events occurs, the system will experience the anomaly described at the top of the tree. Given the specification of a fault tree, minimal cutsets can generally be computed automatically using standard rules of boolean algebra ([**michael2002fault?**](#ref-michael2002fault)).

## 3.7 Exploratory vs. Confirmatory Methods Systems

The anomaly space of the simple linear model system in Figure [1](#modelSLR) is completely determined by the values of or . The specification of the anomaly space here therefore rules out other possible anomalies that may be of interest when applying a linear model. For example, an anomaly such as “One or more regression model residuals are larger than expected when outputted from the model fit" is not contained within the anomaly space of the system, because the regression model residuals are not part of the system output. If there are unusually large residuals from the model fit, they can only cause an anomaly in this system inasmuch as they affect the values of and . Whether this feature is a strength or a weakness of the system depends on the application context, but it suggests that the structure of the anomaly space should be considered when designing a statistical methods system.

### 3.7.1 A Scatterplot+Regression Line System

Suppose that instead of the simple linear model system we used a scatterplot as our statistical method system, with the plot of the data and the overlaid simple linear regression line being the output of interest. This system is shown in Figure [3](#modelPlot).

Scatterplot with overlaid regression line statistical methods system with pseudo-code.

Figure : Scatterplot with overlaid regression line statistical methods system with pseudo-code.

The scatterplot system does not output the estimated regression coefficients (they are intermediate outputs) but rather passes them into a plotting routine that can overlay the fitted regression line.

The scatterplot system in Figure [3](#modelPlot) includes a plot of the data along with the regression line, which is considerably more revealing than the regression coefficients alone. The anomaly space of this system contains at least the following events:

1. Nonlinear structure visible in residuals;
2. Model residuals out of expected range (outliers in the direction);
3. Model leverages out of expected range (outliers in the direction);
4. Intercept or slope of regression line out of expected range

1. .

Note that the anomaly space of the simple linear model system is included within the anomaly space of the scatterplot+regression line system (i.e. case ). Because of the graphical nature of the output, all of the anomaly assessments in the scatterplot+regression line system must be made by eye.

One anomaly we can consider is “A model residual is out of the expected range when viewed on the scatterplot". Figure [4](#ftreePlot) shows the fault tree for the scatterplot+regression line system and this anomaly. Here, we only consider the case where the residual is positive.

Fault tree for unexpected event of “Model residuals out of expected range."

Figure : Fault tree for unexpected event of “Model residuals out of expected range."

Given that a large positive residual implies that , the source of the problem lies with either the data or the model output . Possible problems with the data include errors in the raw data (such as a corrupted file), introduction of error upon reading the data, or some type of measurement problem in the process generating the raw data. Regarding problems with the modeling, it is possible that the model does not capture an important nonlinearity, is missing a key covariate (and the observation is an outlier in that covariate), or that the error structure is incorrectly specified (possibly non-normal or heteroscedastic).

## 3.8 Comparison of Systems

A key difference between the simple linear model system in Figure [1](#modelSLR) and the scatterplot+regression line system in Figure [3](#modelPlot) is the potential size of their respective anomaly spaces. The anomaly space of the simple linear model system is determined by the estimated coefficients from the model while the anomaly space of the scatterplot system is determined by the fitted linear regression line and all of the data points.

The difference between how these two systems can produce anomalies is notably demonstrated in Anscombe’s famous Quartet, which consists of four sets of data on hypothetical variables and . In each of the four datasets, key summary statistics such as the number of observations, Pearson correlation between and , slope of the simple linear regression line, and regression sums of squares are identical ([**anscombe1973graphs?**](#ref-anscombe1973graphs)). If one were to rely on these numerical summaries alone to characterize the data, one would conclude that all of the datasets were the same.

We reproduce Anscombe’s original scatterplots below in Figure [5](#anscombeplot). Upon seeing the four datasets, it is clear that the datasets are not identical and present some features that were perhaps unexpected.

Simple linear regression statistical methods system.

Figure : Simple linear regression statistical methods system.

Each of the scatterplots in Figure [5](#anscombeplot)B–D presents an anomaly if one’s set of expected outcomes specifies that the observed data follow the form under the usual linear model assumptions:

1. Figure Figure [5](#anscombeplot)A does not present an anomaly; it is what one might expect to observe from a simple linear model;
2. Figure Figure [5](#anscombeplot)B is anomalous because the data exhibit a clear nonlinear (possibly quadratic) pattern in addition to a linear trend;
3. Figure Figure [5](#anscombeplot)C is anomalous because the data have an outlier in the -direction;
4. Figure Figure [5](#anscombeplot)D is anomalous because the data have an outlier in the -direction.

In each of Figures [5](#anscombeplot)B–[5](#anscombeplot)D above, the system output (scatterplot+regression line) provides a useful indicator that the simple linear model is inadequate for describing the dataset being plotted.

Anscombe used this quartet of examples as part of an argument for graphing the data versus only computing numerical summaries. The rationale for his argument is not clearly articulated and is along the lines of “We know it when we see it”. But it is possible to formally compare the two systems he proposes (numerical summaries vs. scatterplots) based on an analysis of their anomaly spaces and their fault trees.

The value of the scatterplot+regression line approach in this situation is that its large anomaly space offers many opportunities to interrogate the data analytic process by observing anomalies and question our understanding of the data generation process. When we compare the fault trees in Figure [2](#ftreeSLR) and Figure [4](#ftreePlot) we can see that for the simple linear model, outliers (i.e. large model residuals) represent a root cause of an anomaly related to . However, for the scatterplot+regression line system, large model residuals are a top-level anomaly in and of themselves. Because this top-level anomaly for the scatterplot+regression line system is an “upstream" root cause for the simple linear model, the scatterplot+regression line can trigger an anomaly earlier in the data analytic process than the simple linear model system can.

The difference in the structure of the fault trees in Figures [2](#ftreeSLR) and [4](#ftreePlot) does not indicate that one approach is good and the other is bad, but it does provide a formal characterization of their strengths and weaknesses. Determining which fault tree (and therefore, which system) is more useful depends in part on our confidence in our understanding of the underlying process that generates the data.

### 3.8.1 Characteristics of Exploratory and Confirmatory Systems

The simple linear model system that focuses solely on the regression coefficients is only appropriate in situations where one has decided that the coefficients are the sole quantity of interest and that unusual patterns in the data are tolerable to some degree. This system might be part of a “confirmatory analysis" where problems with the data, such as outliers, are free to occur and are only relevant if they cause an unexpected problem with the estimated coefficients. However, if one is in the phase of data analysis where the data generation process is still being investigated (i.e. ”exploratory analysis") then it would make more sense to choose the scatterplot+regression line system. This system has a large anomaly space and a variety of ways in which anomalies can occur. Therefore, unexpected problems with the data or the modeling can easily be surfaced and possibly rectified by the data analyst.

Characterizing statistical methods systems via their anomaly spaces and fault trees allows us to develop the following generalization with respect to exploratory and confirmatory methods systems:

1. An *exploratory statistical methods system* has a large anomaly space and the fault trees associated with the anomalies have minimal cutsets that are small in size or prioritize single event root causes;
2. A *confirmatory statistical methods system* has a small anomaly space and the fault trees associated with each of the anomalies have minimal cutsets that are large in size and few (if any) single event root causes.

Exploratory systems tend to prioritize visualization or related procedures where the number of outputs is on the order of the sample size itself ([**tukey1977exploratory?**](#ref-tukey1977exploratory); [**cham:clev:1983?**](#ref-cham:clev:1983)). Such approaches provide ample opportunities to detect deviations from expected outcomes in situations where we are learning about the data and the processes underlying them.

Confirmatory systems tend to prioritize robustness and nonparametric flexibility because their outputs are small in number and must be reliable ([**hube:1977?**](#ref-hube:1977); [**tsiatis2007semiparametric?**](#ref-tsiatis2007semiparametric)). Once we have determined which outputs are of interest, we typically want to refine the system to focus on those outputs; increasing the size of the anomaly space may serve as more of a distraction in this type of analysis. Critical to confirmatory statistical methods systems is their ability to protect against failures that are fundamentally unobservable within the context of the data analysis.

A typical pattern that might be followed in going from an exploratory to confirmatory system would be to start with an exploratory system applied to a preliminary dataset. Using a highly iterative process, the exploratory system could reveal a number of anomalies and allow the analyst to reconsider the set of expected outcomes and the design of the statistical methods system. Once the analyst has developed a deeper understanding of the data generation process, a separate (perhaps larger) effort could be made to collect new data. At this point, a confirmatory statistical methods system could be applied to analyze those new data, incorporating the lessons learned from the exploratory analysis.

### 3.8.2 From Exploratory to Confirmatory: “Fixing” Anscombe’s Quartet

Upon seeing the three types of anomalies indicated by the scatterplots in Anscombe’s Quartet (Figure [5](#anscombeplot)B–[5](#anscombeplot)D), we could attempt to revise the simple linear model system in Figure [1](#modelSLR) so that the output would be more reliable and trustworthy when applied to future datasets of this nature. The exploratory analysis based on the scatterplot provided useful information regarding the design of our statistical methods system that we can now incorporate into a revision.

One way we could modify our original system is to design a system that only produces outputs when the data appear as in Figure [5](#anscombeplot)A. That way, we could be reasonably confident that there isn’t significant nonlinear structure or large outliers. Based on our “experience" with Figure [5](#anscombeplot), we can revise the original simple linear model system to handle these anomalies as follows:

1. Read in data
2. Fit model to the data.
3. Fit the alternative model and output . Our expectation is that and (i.e. a simple linear model). If , then the algorithm continues. Otherwise, stop the algorithm.

1. To handle outliers in the -direction, we can compute the leverage for each value from the linear model. If for all (where is the mean of ), then continue.

1. To detect outliers in the -direction we can compute the studentized residuals from the linear model. If for all , then continue.

1. Output and .

This revised algorithm (shown in Figure [6](#modelSLRrev)) introduces three checking steps, each of which has the possibility of stopping the algorithm before completion. The checking steps act as switches—if they evaluate to true, then a a switch is connected allowing the analysis to proceed to the next step. In order for the output to be generated, all three switches in Figure [6](#modelSLRrev) must be closed. Of course, we have made explicit choices here in Steps – that might have been different depending on the circumstances of the analysis.

Revised simple linear regression system.

Figure : Revised simple linear regression system.

One might recognize the three additional checks presented in Steps – as common checks of the assumptions of the linear model. Indeed, “checking the assumptions” for a linear regression model can be thought of as a way to decrease the size of the anomaly space for a statistical methods system while also ensuring that erroneous or misleading results do not emerge from the system. Alternatively, we can say that these checks increase the size of the minimal cutsets for the fault trees in the anomaly space.

Consider the anomaly “ when outputted from the model fit" for the system in Figure [6](#modelSLRrev). The fault tree for the revised simple linear regression system with additional checks is shown in Figure [7](#ftreeSLRrev). The top of the fault tree for this anomaly class is similar to the one shown in Figure [2](#ftreeSLR). However, when we take the branch that considers”contaminated data", things begin to change. We can see that the two AND gates ( and ) below the event “Contaminated data produce slope " imply that multiple events have to occur in order to trigger this type of anomaly.

Revised fault tree for simple linear regression system with additional checks.

Figure : Revised fault tree for simple linear regression system with additional checks.

The minimal cutsets for this part of the fault tree are all of size 5, meaning that five events have to occur in order to trigger the anomaly at the top of the tree. For example, one of the cutsets indicates that if contaminated data were to produce , we would require that (1) outliers were present (); and (2) the contamination was arranged to produce (); and (3) the quadratic coefficient in the quadratic model was close to 0 (); and (4) all residuals (); and (5) all leverages (). While it is possible to imagine this scenario occurring as a result of random fluctuations in the data, it is more likely that we would observe this pattern because of a change in the process governing the relationship between and (i.e. the other branch at the top of the tree). In other words, with this revised statistical methods system, if the anomaly at the top of the tree were observed to occur, the more likely explanation is that our model is incorrect rather than there being a problem with the data.

## 3.9 Discussion

Data analysis is often depicted as an iterative process, but the precise details of how this iteration works have not been specified in a manner that can be broadly generalized. This paper attempts to systematically describe this iterative process by framing a critical aspect of data analysis as a process involving the recognition of anomalies and the enumeration of their root causes. The consideration of anomalies in data analysis provides a mechanism for interpreting statistical output in a manner that allows us to adjust our expectations about the data generation process and revise the statistical methods systems we build.

Perhaps the most useful aspect of framing the data analytic process in terms of anomalies is that it forces us to intentionally consider a set of expected outcomes when a statistical methods system is applied to data. This “pre-registration" of outcomes is what allows us to detect anomalies in the first place and gives us a concrete basis for interpreting the output from an analysis. While experienced analysts likely already do this (perhaps unconsciously) and can iterate through a series of anomalies and root causes quickly, having an explicit statement of expected outcomes and a formal methodology for identifying root causes may be valuable to novices who are still learning the craft. Some have suggested that the reliance on explicit hypotheses can lead analysts to overlook important aspects of the data ([**yanai2020hypothesis?**](#ref-yanai2020hypothesis)). However, we would argue that it is not the pre-specification of outcomes that is the issue here, but rather the choice of confirmatory methods systems in situations where the data generating process is not well-understood.

We have proposed the use of fault trees as a formal approach to identifying the possible root causes of a data analytic anomaly. This methodology is well-established in the systems engineering literature and its general properties are described elsewhere ([**michael2002fault?**](#ref-michael2002fault)). However, we struggle to find any explicit use of fault trees in the statistical analysis literature. We believe fault trees have something to contribute to the process of doing data analysis. They force analysts to consider the anomaly space of a statistical methods system and to interrogate the various possible anomalies that the system may produce. Fault trees provide a new approach to comparing the strengths and weaknesses of different analytic systems in terms of how anomalies can be identified and under what conditions they might occur. In particular, they provide a way to characterize the differences between exploratory and confirmatory statistical methods systems. Finally, fault trees can record the historical knowledge of specific types of data analysis and can serve as a useful educational tool for teaching newcomers to those areas.

We hypothesize that the breadth and depth of a fault tree are directly connected to the range of experience of the analyst or analysts who are using the system. Experience can be gained over time or by working with and learning from a wide range of collaborators. Diversity of experience, both across time and across people, can only serve to improve the quality and usefulness of the fault tree. Furthermore, the fault tree can be passed down as a record of experience to new analysts working with the same system or with similar data.

A topic not covered here is the question of how can analysts recognize an anomaly in the first place? Anomaly *detection* requires having enough background knowledge to create a reasonable set of expected outcomes. If the set of expected outcomes is too large, then there will be no chance of observing any anomalies and no way to inform our expectations or update our analytic systems. The ability to construct a proper set of expected outcomes can be reinforced through rigorous preparation (e.g. summarizing the literature, meta-analysis), pre-registration of hypotheses ([**asendorpf2013recommendations?**](#ref-asendorpf2013recommendations)), and experience working with similar data.

Data analysis is a process that involves making a series of decisions and the manner in which we make those decisions is often not well-documented ([**simmons2011false?**](#ref-simmons2011false); [**gelman2014statistical?**](#ref-gelman2014statistical)). While fault trees provide a formal mechanism for enumerating the possible root causes of an anomaly, the analyst must then decide what to do next. For example, the analyst must decide which collection of root causes should be further investigated. This choice could be made based on the analyst’s determination of the relative likelihood of each of the root causes, with the most likely cause being the highest priority. Given the decision-making nature of data analysis, it may be possible to employ the tools of decision theory to optimize the analyst’s choices ([**smelser2001international?**](#ref-smelser2001international)). However, any application of decision theoretic tools would require a much more precise specification of the decision process than is likely available. Notions of utility and consequences may only be vaguely known and it may not be possible to assign a probability distribution to the various root causes in the fault tree. However, with experience, and perhaps after collection of some data on decisions made in the past, it may be possible to employ a formal decision-theoretic framework to the data analytic decision-making process.

In this paper we have intentionally avoided discussion of software implementations. Modern data analyses are necessarily implemented in software, which can produce their own anomalies separately for reasons that may or may not be related to the data analysis itself. For example, a poorly formatted data file might cause software reading in that data file to crash. Data analysts must to some extent be able to trace anomalies or outright failures to possible software-related root causes. Therefore, familiarity with software implementations may be of equal importance to familiarity with the statistical properties of the methods implemented. Our discussion of anomalies here parallels ideas in software unit testing, which is a practice that is employed to ensure that software anomalies are detected in the development process ([**runeson2006survey?**](#ref-runeson2006survey); [**testthat2011?**](#ref-testthat2011)).

Considering the root causes of data analytic anomalies reinforces the importance of the reproducibility of analyses ([**peng2011reproducible?**](#ref-peng2011reproducible)). Reproducible research allows others to reconstruct the statistical methods system that was applied to the data. In a “post-mortem" situation where an anomaly is being investigated, possibly by an independent third party, having the code and datasets available is critical for reconstructing the sequence of events that lead to the anomaly. Without such information, it is impossible to construct the fault tree necessary for enumerating the possible root causes.

It is interesting to consider the approach proposed here in the context of modern machine learning algorithms. Iterative methods like boosting essentially attempt to automate the evaluation of anomalies and update their predictions successively based on pre-determined rules for evaluating their fault trees. The original AdaBoost algorithm re-weighted misclassified observations more heavily so that successive iterations would produce weak classifiers focused on those values ([**friedman2002stochastic?**](#ref-friedman2002stochastic); [**friedman2000additive?**](#ref-friedman2000additive)). This implies that in evaluating the anomaly of a misclassified observation, AdaBoost always takes the branch of the fault tree that considers the model to be somehow incorrect. Many have pointed out that the performance of such algorithms is degraded by outliers and have proposed robust alternatives ([**long2010random?**](#ref-long2010random); [**li2018boosting?**](#ref-li2018boosting)).

In this paper we focus on what can be learned from a single dataset in a given data analysis. Our goal was to develop an approach that would be useful to a data analyst at the moment of data analysis. However, there clearly are limits on what can be learned from the observed data. As presented here, some anomalies, such as outliers or nonlinear structure, can be observed. But other problems, such as the presence of unobserved confounders, that might affect the external validity of an analysis, simply cannot be checked with the observed data ([**ogburn2019comment?**](#ref-ogburn2019comment)). Hence, we must draw a distinction between “assumptions" that can be checked and genuine assumptions that simply must be assumed to be true. In general, checkable assumptions can be evaluated with the observed data and any information gained can be incorporated into the statistical methods system (as we demonstrated in Section [5.4](#sec:slrfixed)).

The process of doing data analysis is complex and often idiosyncratic but statisticians should not refrain from attempting to describe it formally. As data science and statistics become more fundamental to many aspects of society, we need to develop ways to generalize our approach to analyzing data and translate those ideas to ever wider audiences. We propose that characterizing data analysis as a process of diagnosing anomalies and identifying root causes via fault trees is one such generalizable approach that can serve as a guide in a wide variety of data analyses.

# 4 Fault Tree Notation

Comprehensive details on constructing fault trees can be found in Vesely, et al. ([**vesely1981fault?**](#ref-vesely1981fault)) and Ericson ([**ericson2011fault?**](#ref-ericson2011fault)). We provide an abbreviated summary here that is sufficient for understanding the fault trees in this paper.

If we consider the system shown in Figure [1](#modelSLR), we could implement it with the following R code.

## Read in data from CSV file  
dat <- read.csv("dataset.csv")   
  
## Fit linear model  
fit <- lm(y ~ x, data = dat)  
  
## Extract estimated coefficients  
b <- coef(fit)  
  
## Return `b'

If we assume that the code underlying the execution of the system works as expected and that the data are properly formatted and clean, then the code will execute and coefficient vector b will be returned. If we are to consider what are the failure modes of this system, it must be noted that the coefficient vector b is the *only* output from the system according to Figure [1](#modelSLR). Therefore, any failure modes associated with the system must correspond to the coefficient values.

# 5 Analysis of Alternatives

## Learning Objectives

The learning objectives of this chapter are:

1. Draw a block diagram describing a data analysis
2. Describe alternatives to elements of a data analysis block diagram

## 5.1 Narrative Structure

Upon reading about or perhaps building a data analysis, it is important that we have a formal approach to critiquing the analysis and proposing improvements.

## 5.2 Block Diagrams

## 5.3 Specifying Alternatives

## 5.4 Analysis of Alternatives

Once alternative approaches have been specified, it is important to consider the evidence for or against each alternative.

# About the Authors

These credits are based on our [course contributors table guidelines](https://www.ottrproject.org/more_features.html#giving-credits-to-contributors).

| Credits | Names |
| --- | --- |
| **Pedagogy** |  |
| Lead Content Instructor(s) | [Roger Peng] |
| Lecturer(s) (include chapter name/link in parentheses if only for specific chapters) - make new line if more than one chapter involved | Delivered the course in some way - video or audio |
| Content Author(s) (include chapter name/link in parentheses if only for specific chapters) - make new line if more than one chapter involved | If any other authors besides lead instructor |
| Content Contributor(s) (include section name/link in parentheses) - make new line if more than one section involved | Wrote less than a chapter |
| Content Editor(s)/Reviewer(s) | Checked your content |
| Content Director(s) | Helped guide the content direction |
| Content Consultants (include chapter name/link in parentheses or word “General”) - make new line if more than one chapter involved | Gave high level advice on content |
| Acknowledgments | Gave small assistance to content but not to the level of consulting |
| **Production** |  |
| Content Publisher(s) | Helped with publishing platform |
| Content Publishing Reviewer(s) | Reviewed overall content and aesthetics on publishing platform |
| **Technical** |  |
| Course Publishing Engineer(s) | Helped with the code for the technical aspects related to the specific course generation |
| Template Publishing Engineers | [Candace Savonen](https://www.cansavvy.com/), [Carrie Wright](https://carriewright11.github.io/) |
| Publishing Maintenance Engineer | [Candace Savonen](https://www.cansavvy.com/) |
| Technical Publishing Stylists | [Carrie Wright](https://carriewright11.github.io/), [Candace Savonen](https://www.cansavvy.com/) |
| Package Developers ([ottrpal](https://github.com/jhudsl/ottrpal)) [Candace Savonen](https://www.cansavvy.com/), [John Muschelli](https://johnmuschelli.com/), [Carrie Wright](https://carriewright11.github.io/) |  |
| **Art and Design** |  |
| Illustrator(s) | Created graphics for the course |
| Figure Artist(s) | Created figures/plots for course |
| Videographer(s) | Filmed videos |
| Videography Editor(s) | Edited film |
| Audiographer(s) | Recorded audio |
| Audiography Editor(s) | Edited audio recordings |
| **Funding** |  |
| Funder(s) | Institution/individual who funded course including grant number |
| Funding Staff | Staff members who help with funding |

## ─ Session info ───────────────────────────────────────────────────────────────  
## setting value   
## version R version 4.0.2 (2020-06-22)  
## os Ubuntu 20.04.5 LTS   
## system x86\_64, linux-gnu   
## ui X11   
## language (EN)   
## collate en\_US.UTF-8   
## ctype en\_US.UTF-8   
## tz Etc/UTC   
## date 2023-03-23   
##   
## ─ Packages ───────────────────────────────────────────────────────────────────  
## package \* version date lib source   
## assertthat 0.2.1 2019-03-21 [1] RSPM (R 4.0.5)   
## bookdown \* 0.24 2023-03-10 [1] Github (rstudio/bookdown@88bc4ea)   
## cachem 1.0.7 2023-02-24 [1] CRAN (R 4.0.2)   
## callr 3.5.0 2020-10-08 [1] RSPM (R 4.0.2)   
## cli 3.6.0 2023-01-09 [1] CRAN (R 4.0.2)   
## crayon 1.3.4 2017-09-16 [1] RSPM (R 4.0.0)   
## desc 1.2.0 2018-05-01 [1] RSPM (R 4.0.3)   
## devtools 2.3.2 2020-09-18 [1] RSPM (R 4.0.3)   
## digest 0.6.25 2020-02-23 [1] RSPM (R 4.0.0)   
## ellipsis 0.3.1 2020-05-15 [1] RSPM (R 4.0.3)   
## evaluate 0.20 2023-01-17 [1] CRAN (R 4.0.2)   
## fastmap 1.1.1 2023-02-24 [1] CRAN (R 4.0.2)   
## fs 1.5.0 2020-07-31 [1] RSPM (R 4.0.3)   
## glue 1.4.2 2020-08-27 [1] RSPM (R 4.0.5)   
## htmltools 0.5.4 2022-12-07 [1] CRAN (R 4.0.2)   
## knitr 1.33 2023-03-10 [1] Github (yihui/knitr@a1052d1)   
## magrittr 2.0.3 2022-03-30 [1] CRAN (R 4.0.2)   
## memoise 2.0.1 2021-11-26 [1] CRAN (R 4.0.2)   
## pkgbuild 1.1.0 2020-07-13 [1] RSPM (R 4.0.2)   
## pkgload 1.1.0 2020-05-29 [1] RSPM (R 4.0.3)   
## prettyunits 1.1.1 2020-01-24 [1] RSPM (R 4.0.3)   
## processx 3.4.4 2020-09-03 [1] RSPM (R 4.0.2)   
## ps 1.4.0 2020-10-07 [1] RSPM (R 4.0.2)   
## R6 2.4.1 2019-11-12 [1] RSPM (R 4.0.0)   
## remotes 2.2.0 2020-07-21 [1] RSPM (R 4.0.3)   
## rlang 1.0.6 2022-09-24 [1] CRAN (R 4.0.2)   
## rmarkdown 2.10 2023-03-10 [1] Github (rstudio/rmarkdown@02d3c25)  
## rprojroot 2.0.3 2022-04-02 [1] CRAN (R 4.0.2)   
## sessioninfo 1.1.1 2018-11-05 [1] RSPM (R 4.0.3)   
## stringi 1.5.3 2020-09-09 [1] RSPM (R 4.0.3)   
## stringr 1.4.0 2019-02-10 [1] RSPM (R 4.0.3)   
## testthat 3.0.1 2023-03-10 [1] Github (R-lib/testthat@e99155a)   
## usethis 1.6.3 2020-09-17 [1] RSPM (R 4.0.2)   
## withr 2.3.0 2020-09-22 [1] RSPM (R 4.0.2)   
## xfun 0.26 2023-03-10 [1] Github (yihui/xfun@74c2a66)   
## yaml 2.2.1 2020-02-01 [1] RSPM (R 4.0.3)   
##   
## [1] /usr/local/lib/R/site-library  
## [2] /usr/local/lib/R/library

# 6 References